Low Rank Approximation

Idea: Given a matrix A, find a smaller rank matrix that best approximates A. This is used for many purposes:
Image Processing / Compression, Data Mining
Machine Learning, Noise reduction (in data), etc.

First note that for nonzero vectors $u \in \mathbb{R}^m$ and $v \in \mathbb{R}^n$, uv^T is an $m \times n$ matrix of rank one:

$$\begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} \begin{bmatrix} v_1 \cdots v_n \end{bmatrix} = \begin{bmatrix} u_1v_1 & u_1v_2 & \cdots & u_1v_n \\ u_2v_1 & u_2v_2 & \cdots & u_2v_n \\ \vdots & & & \\ u_mv_1 & - & - & u_mv_n \end{bmatrix}$$

Rank 1 since each row is a multiple of v^T.

Moreover, rank r matrices can be expressed as the sum of r rank one matrices:

Ex:
$$\begin{pmatrix} 1 & 2 & 0 & 4 \\ 2 & 4 & 0 & 8 \\ 0 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 0 & 4 \\ 2 & 4 & 0 & 8 \\ 0 & 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix}$$

rank 2 — rank 1 — rank 1

$$= \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix} (1204) + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} (0111)$$ ← less data

In general, if A has rank r, then $A = u_1v_1^T + \cdots + u_rv_r^T$ where $u_1, \ldots, u_r \in \mathbb{R}^m$ and $v_1, \ldots, v_r \in \mathbb{R}^n$

The SVD of A can be expressed as the sum of $r = \text{rank}(A)$ rank one matrices:

$$A = U\Sigma V^T = [u_1 \cdots u_m] \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_r \end{bmatrix} [v_1 \cdots v_n]^T$$

$$= [\sigma_1 u_1 \cdots \sigma_r u_r \, 0 \cdots 0] \begin{bmatrix} v_1^T \\ \vdots \\ v_n^T \end{bmatrix}$$

$$= \sigma_1 u_1 v_1^T + \cdots + \sigma_r u_r v_r^T$$

Def: The SVD rank k approximation of A is

$$A_k := \sigma_1 u_1 v_1^T + \cdots + \sigma_k u_k v_k^T$$

Note: $\sigma_1, \ldots, \sigma_k$ are the k largest singular values.

Notation: $A_k = \underbrace{[u_1 \cdots u_k]}_{U_k} \underbrace{\begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix}}_{\Sigma_k} \underbrace{\begin{bmatrix} v_1^T \\ \vdots \\ v_k^T \end{bmatrix}}_{V_k}$ and $A_r = A$

Ex: $A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 2/\sqrt{6} & 0 & -1/\sqrt{3} \\ 1/\sqrt{6} & 1/\sqrt{2} & -1/\sqrt{3} \\ 1/\sqrt{6} & -1/\sqrt{2} & -1/\sqrt{3} \end{pmatrix} \begin{pmatrix} \sqrt{3} & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix}$

The rank one approximation of A is

$$A_1 = \sqrt{3} \begin{pmatrix} 2/\sqrt{6} \\ 1/\sqrt{6} \\ 1/\sqrt{6} \end{pmatrix} \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix} \quad \text{and} \quad A_2 = A$$

Key Idea: If $\sigma_{k+1}, \ldots, \sigma_r$ are close to 0, then A_k gives a good approximation of A

This isn't the only low rank approximation, but it is the best one when measured using the Frobenius norm.

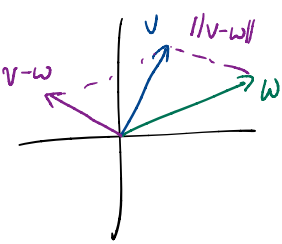
Def: The Frobenius norm of a matrix A is $\|A\|_F = \sqrt{\sum_{i,j} a_{ij}^2}$

Ex: $\left\| \begin{pmatrix} 1 & -1 \\ 1 & 0 \\ 2 & 1 \end{pmatrix} \right\|_F = \sqrt{1^2+1^2+2^2+(-1)^2+0^2+1^2} = \sqrt{8}$

Ex: If A is $n \times 1$ (ie. A is a vector), then $\|A\|_F$ is just the length of A, as a vector.

The Frobenius distance between $m \times n$ matrices A and B is $\|A-B\|_F$.

Ex: If $v, w \in \mathbb{R}^n$, then $\|v-w\|_F$ is just the usual distance between v and w



We use the Frobenius distance to measure the quality of an approximation. If $\|A-B\|_F$ is small, then B is a good approximation of A.

Theorem: A_k is the best rank k approximation of A when using the Frobenius norm

(ie. if B is another rank k approximation, then $\|A-B\|_F \geq \|A-A_k\|_F$)

Proof:

Let $A = \begin{bmatrix} x_1^T \\ \vdots \\ x_m^T \end{bmatrix}$

First note that every rank k $m \times n$ matrix B gives rise to a k-dim'l subspace of $\mathbb{R}^n$, $C(B^T)$

Fix a k-dim'l subspace W and

Let $B = \begin{bmatrix} b_1^T \\ \vdots \\ b_m^T \end{bmatrix}$ satisfy $C(B^T) = W$.

We fixed decide which such B minimizes $\|A-B\|_F$.

Notice $\|A-B\|_F^2 = \left\| \begin{bmatrix} b_1^T - x_1^T \\ \vdots \\ b_m^T - x_m^T \end{bmatrix} \right\|_F^2 = \sum_{i=1}^m \|b_i - x_i\|^2$

is the sum of the square distances between x_i and $b_i \in W$. this quantity is minimized when $b_i = \text{proj}_W x_i$

So for a fixed k-dim'l subspace W,

$\|A-B\|_F^2$ is minimized when $B = \begin{bmatrix} (\text{proj}_W x_1)^T \\ \vdots \\ (\text{proj}_W x_m)^T \end{bmatrix}$

(where B is a matrix with $C(B) = W$)

In this case, $\|A-B\|_F^2 = \sum_{i=1}^m \|\text{proj}_W b_i - x_i\| = \sum_{i=1}^m \|e_i\|^2$

Now which k-dim'l subspace minimizes $\sum \|e_i\|^2$?
The k-dim'l subspace of best fit W'!

Recall that W' has an orthonormal basis given by $\{v_1, \dots v_k\}$ where $A = U\Sigma[v_1 \cdots v_n]^T$ is an SVD of A.
So $\|A-B\|_F^2$ is minimized when $B = \begin{bmatrix} \text{proj}_{W'} x_1 \\ \vdots \\ \text{proj}_{W'} x_m \end{bmatrix}$.

We must determine $\text{proj}_{W'} x_i \quad \forall\, i$.

Note: x_i^T is the ith row of $A = U\Sigma V^T$

which is given by $x_i^T = \sigma_1 u_{i1} v_1^T + \cdots + \sigma_r u_{ir} v_r^T$

Hence

$$\text{proj}_{W'} x_i = \sum_{j=1}^{k} \text{proj}_{v_j} x_i = \sum_{j=1}^{k} (v_j \cdot x_i) v_j$$

$$= \sum_{j=1}^{k} (v_j \cdot (\sigma_1 u_{i1} v_1 + \cdots + \sigma_r u_{ir} v_r)) v_j$$

$$= \sum_{j=1}^{k} (\sigma_j u_{ij}) v_j \qquad \text{Since } v_j \cdot v_i = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}$$

$$= \sigma_1 u_{i1} v_j + \cdots + \sigma_k u_{ik} v_k$$

$$= i\text{th row of } A_k!$$

Hence A_k minimizes $\|A-B\|_F$ ∎